Christian-Albrechts-Universität zu Kiel
Institut für Informatik
Prof. Dr. W.-P. de Roever
Jens Schönborn und Jan Waller

Nebenläufige und verteilte Programmierung

Wintersemester 2006/07

Serie 10

8.1. 2007

Ausgabetermin: 8.1. 2007

Abgabe: 15.1. 2007 (11:00)

Aufgabe 1 (2 Punkte) Betrachte den self-scheduling disk driver Prozess in Abblidung 7.9 auf Seite 311.

1. Modifiziere den Prozess so, dass die CSCAN scheduling Strategie (siehe Seite 227) verwendet wird. Pseudocode genügt. Kommentiere und erkläre die Lösung.

2. Modifiziere den Prozess so, dass die SCAN scheduling Strategie (siehe Seite 227) verwendet wird. Pseudocode genügt. Kommentiere und erkläre die Lösung.

Aufgabe 2 (4 Punkte) Consider the program in Figure 7.7. Answer the following as independent exercises.

1. Modify the program to allow clients to request and release more than one unit at a time. Give comments and explanation for your solution.

2. The program shows how to simulate the specific monitor in Figure 7.6 using a server process. Develop a simulation of an arbitrary monitor. In particular, show how you would simulate every monitor mechanism in Section 5.1, i.e. wait, priority wait, signal, signal_all, empty, and minrank. Give pseudocode for both the client processes as well as for the server. Give comments and explanation (especially the interaction between the clients and the server) for your code.

Aufgabe 3 (4 Punkte) Linda's tuple space could be implemented by a server process. Application processes would then interact with the server whenever they wanted to execute a Linda primitive such as OUT, IN, or EVAL. Develop pseudocode for the server process showing how it could be implement tuple space and the six Linda primitives OUT, IN, RD, INP, RDP, and EVAL. Also show the (pseudo)code that an application process would execute for each primitive. The application and server processes should interact using asynchronous message passing. Give comments and explanation for your solution.